ImRec: Learning Reciprocal Preferences Using Images

JAMES NEVE, Univeristy of Bristol, United Kingdom
RYAN MCCONVILLE, Univeristy of Bristol, United Kingdom

Reciprocal Recommender Systems are recommender systems for social platforms that connect people to people. They are commonly used in online dating, social networks and recruitment services. The main difference between these and conventional user-item recommenders that might be found on, for example, a shopping service, is that they must consider the interests of both parties. In this study, we present a novel method of making reciprocal recommendations based on image data. Given a user's history of positive and negative preference expressions on other users images, we train a siamese network to identify images that fit a user's personal preferences. We provide an algorithm to interpret those individual preference indicators into a single reciprocal preference relation. Our evaluation was performed on a large real-world dataset provided by a popular online dating service. Based on this, our service significantly improves on previous state-of-the-art content-based solutions, and also out-performs collaborative filtering solutions in cold-start situations. The success of this model provides empirical evidence for the high importance of images in online dating.

CCS Concepts: • **Information systems** → **Social recommendation**; *Decision support systems*; *Personalization*; • **Computer systems organization** → Neural networks.

Additional Key Words and Phrases: Reciprocal Recommender Systems, Content-Based Recommendation, Siamese Networks, Social Recommendation

ACM Reference Format:
James Neve and Ryan McConville. 2020. ImRec: Learning Reciprocal Preferences Using Images. In *Fourteenth ACM Conference on Recommender Systems (RecSys '20), September 22–26, 2020, Virtual Event, Brazil.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3383313.3411476

1 INTRODUCTION

Reciprocal Recommender Systems (RRS) are a subtype of Recommender Systems (RS) that recommend people to people [19]. They are primarily used in social services such as online dating, social networking and recruitment. In spite of the importance of these areas to society, RRSs have received relatively little attention in the literature compared to the traditional non-reciprocal RS setting, such as movie and shopping recommendation.

RRSs are inherently complex because they must consider the preferences of two users for each other, as compared to conventional RSs, which estimate a user's preference for an inanimate item. Success in the RRS setting is making a recommendation where both the user seeing the recommendation and the user being recommended are happy with the match. This bidirectional preference relation is often not balanced, with concepts such as user popularity and how active or passive the user is playing a role in the value of their preference [2]. Most RRSs in the literature are evaluated on data from online dating, which provides clear binary signals of attraction, but RRSs for social services [23] and job recruitment [36][31] have also been developed.

RecSys '20, September 22–26, 2020, Virtual Event, Brazil

ACM ISBN 978-1-4503-7583-2/20/09…$15.00
https://doi.org/10.1145/3383313.3411476

There are a number of examples of collaborative filtering-based RRSs in the literature [26][22]. However, besides RECON [20], a categorical content-based RRS, there are very few examples of content-based reciprocal recommendation. Social networks and in particular online dating are content-rich fields, with users prepared to spend much more time writing personal profiles and selecting images than they would be on, for example, a shopping service. Content-based RSs, while often outperformed by collaborative filtering algorithms in data-rich environments [5], tend to perform better in cold-start situations [33][15] and often improve the results of collaborative filtering as part of hybrid systems [6]. It is therefore necessary to keep innovating in this area and improve on the state of the art.

Content-based RRSs in the literature have only been designed to work with categorical data. RECON was trained on a service that did not include user image data. This type of service is in the minority - most online dating services use images, and some very popular ones such as *Tinder*[1] encourage users to make initial decisions based entirely on images. There is informal evidence that users on dating services overwhelmingly make decisions based on image data, even when detailed text profiles are available[2]. In addition, recent social networks such as *Instagram*[3] often focus on images rather than written or categorical content. As such, attractiveness of users, generalised to attractiveness of images to individuals, could be used to improve social RRSs. As attractiveness is subjective, this measure should account for the tastes of individual users, and make recommendations based on specifically who is attractive to whom. In this paper, we use the pharse "personal attractiveness" to refer to the attractiveness of one person's image to another person. We consider that the whole image is potentially a trigger for attraction, and not only the person in the image's physical appearance.

To overcome the limitations of content-based RRSs that use only categorical data, we present an original recommender system, *ImRec* that predicts preference of users x and y for each other given their images and their history of positive and negative preferences. ImRec is based on a Siamese Neural Network that predicts, given an image of a user already liked by x and an image of y, whether x will like y. To the best of our knowledge, this is the first example in the literature of a machine learning model successfully predicting personal attractiveness. The results from this model are aggregated across the history of x and y's preferences, to give two unidirectional preference scores. These two scores are then aggregated into a single bidirectional preference relation. We demonstrate, through offline testing, that the model is capable of successfully differentiating between positive and negative indicators of preference in this context, where the baseline algorithm RECON was not able to.

This paper was produced in collaboration with *Pairs*[4], the online dating service in Japan with the largest number of registered users. Pairs is marketed to people searching for a long-term serious relationship. The service has approximately 10 million registered users, and we used a dataset of 250000 users and 1 million positive and negative expressions of preference in our model, taken from the most recent users and interactions on the service. Users on Pairs must verify their identity with a photographic ID card such as a driving license. Images posted by users on Pairs are mostly photos, and are all manually checked to ensure that the user and no other people are present in the photo. We therefore consider our dataset of relatively high quality.

This paper's contribution is threefold:

- We are the first, to the best of our knowledge, to propose a model that predicts personal attractiveness based on image data and positive and negative expressions of preference. This work is also the first evidence-based demonstration of the high importance of images in online dating.

[1] https://tinder.com/
[2] https://www.gwern.net/docs/psychology/okcupid/weexperimentonhumanbeings.html
[3] https://www.instagram.com/
[4] https://pairs.lv

- We design an RRS, ImRec, based on our deep learning model, training a random forest to combine individual image preference scores into two unidirectional preference scores, which are subsequently combined into a single bidirectional preference relation.
- We demonstrate that our model significantly outperforms the previous baseline on a very large real world dataset consisting of 300000 images and 1 million preferences, and postulate that this performance is likely to generalise to any setting where RRSs are used and image data is available.

2 RELATED WORKS

This section provides an overview of literature that forms the background for this paper. This includes papers in the fields of content-based recommendation, reciprocal recommendation, and machine learning models for attractiveness.

2.1 Reciprocal Recommendation

RRSs aim to create a bidirectional matching between two entities [19]. This makes them inherently more complex than user-item RSs, which only need to consider the preferences of a single user. They are commonly used in online dating [20], recruitment [36] and social networks [12].

The earliest and most recent example of an RRS in the literature is *RECON* [20]. RECON uses implicit preferences to make recommendations. Implicit preferences are inferred from a user's message history. If a user x sends messages to users y_1, y_2 and y_3, attributes that those users have in common will be considered preferences of x.

In order to calculate the bidirectional preference score between x and y, RECON calculates $Q_{x,y}$ and $Q_{y,x}$ and then takes the harmonic mean of the two values. RECON was more successful than the baselines of unidirectional collaborative filtering and user search.

Following RECON, the first RRS based on collaborative filtering was *Reciprocal Collaborative Filtering* (RCF) [26]. For candidate users x and y, RECON calculates the similarity between x and other users who have liked y, and the similarity between y and other users who have liked x, to get two preference scores. These preference scores are then aggregated using the harmonic mean to get a bidirectional preference relation, which can be used to make recommendations. RCF improved on the results of RECON, and also introduced a modification for RECON where it was used as a baseline, where continuous attributes are calculated as a distance metric instead of being binned, improving the accuracy for users at the edge of bins. We use this modification in our implementation of RECON as a baseline.

Since the introduction of RCF, a number of improvements to the system, or alternative hybrid RRSs have been designed. Kleinerman et al. designed a modification to RCF to account for user popularity [2] and also tested explanations for the recommendations made by RCF, finding that users provided with explanations for their recommendations were more likely to use them [3]. Neve et al. designed a collaborative filtering algorithm based on latent factors, and found that this improved on the efficiency of RCF on large datasets [22]. There has also been research performed on alternative aggregation operators to the harmonic mean for combining unidirectional preference scores into a bidirectional relation, finding that depending on the situation, different operators were likely to be more or less appropriate [21].

2.2 Content-Based Recommendation

Content-based RSs make recommendations based on user preferences. These preferences are either explicitly input by the user, or implicitly inferred based on the user's history of preferences [1]. Content-based recommender systems that operate entirely on preferences for categorical data, such as RECON, are algorithmically simple and tend not to appear in the literature.

Making recommendations using unstructured data, such as text, images, sound and videos, is a more challenging problem. Often the problem is solved by representing the data as an embedding that can then be compared to other embeddings to produce a similarity metric [13]. In particular, news recommendation is a very rich field, and early examples used simple metrics such as *tf-idf* to identify the similarity between previously liked text documents for news recommendation [16][30]. More recent approaches such as [29] often use deep learning approaches to generate these embeddings.

There are comparatively few examples of content-based RSs that use images to make recommendations. Lei et al. used positive and negative preferences for images to train a model based on ImageNet [14] that predicts user preference for one of two images [7]. *DeepStyle* [27] used a *Comparative Network* to predict user preference for clothes based on image data. However, given the richness of image data and the obvious impact of images on user preferences, there is scope for significantly more research in this area. In particular, the datasets used in these tests were carefully curated public datasets. The data for our algorithm comprised of user submitted images, which are often low quality or have filters pre-applied to them. Based on our tests, the former of the above methods did not work well, although some aspects of our model including the siamese network we used is similar to the Comparative Network used in the latter.

2.3 Machine Learning for Attractiveness

There has been extensive research on identifying features of images using machine learning. Most recent successful attempts to identify arbitrary objects in images have used Convolutional Neural Networks (CNN), for example [14][8]. CNNs also perform exceptionally well on more specific tasks such as identifying people within images [9]. Because of results such as these on images similar to those in our dataset, we hypothesized that in our case, a CNN would be a promising approach for extracting relevant features from user images for estimating attractiveness.

To the best of our knowledge, there are no examples in the literature of papers identifying *personal attractiveness* (i.e. how attractive is y's image to user x) using machine learning. There are small number of examples of machine learning models to identify *general attractiveness* (i.e. how attractive is y to the average potential partner) based on image data [32]. This was a small study, using a non-machine learning model, and based on the opinions of a small group of recruited students. There are also models that claim to identify a user's physical attributes such as age and even height based on face images [28][34].

As none of these models that identify attractiveness or physical features report the sizes of their classes, it is difficult to know how well the models perform against the baseline of guessing the average. For example, [32] was a small initial study and missing some crucial information which would allow us to draw conclusions about its effectiveness. We therefore conclude that our model, which identifies personal attractiveness as part of a large controlled study, is unique in the literature.

3 METHODOLOGY

In this section, we describe a model that predicts user preference for images.

3.1 Data

The data for our model was provided by *Pairs*, a Japanese dating service with approximately 10 million users registered in Japan. Before they can exchange messages or contact details, users on Pairs must interact with each other using binary indicators of preference known as *Likes*. Users initially find others by viewing list pages where they can see a variety of other users' main images, nicknames and ages. After clicking on a user's icon, they can see images, categorical data such as age and body type, and detailed text profiles.

If a user x decides that they want to interact with another user y, x sends y a *Like*. User y is alerted to this, and has the choice to respond with either a *Like* or a *Nope*. If both users *Like* each other, this is considered a *Match* and the users can then exchange messages. Free users are initially given 30 *Likes*, and users who subsequently subscribe to the service are given a further 30 per month. This encourages users to be discerning and not *Like* in great quantity in the hopes of getting any response.

In our model, we use *Likes* and *Nopes* as positive and negative indicators of preference respectively. In order to build a model that differentiates between positive and negative user preferences for images, for each user x we sampled training data in triplets of one anchor user *Liked* by x, y_a, one other *Liked* user y_p and one *Noped* user y_n. For y_a, y_p and y_n, we used their main image under the assumption that, as the largest image and the one that appears on list pages, it is the most important for users making their decision about whether to indicate positive or negative preference.

Users sometimes post photos or images in which they are not included, such as photos of their favourite food or scenery. While these images may have an impact on attraction, we wanted to focus on users' personal attraction to each other. We therefore used the Python wrapper for the Dlib [5] library to identify images with human faces, and exclude those where no faces were detected. The vast majority of users supply at least one photo in which their face is shown, but to ensure fairness for users that do not, alternative recommendation algorithms would need to be applied for their case.

We sampled triplets of images from 250000 users, and this sampling resulted in 500000 triplets of y_a, y_p and y_n images. In order to maintain generalisability and ensure that the model was not biased to any specific user, we did not sample more than 10 triplets from any specific user. In order to further increase the reliability of the system, we excluded users who had not verified their identity by sending a copy of photographic ID that was checked by *Pairs* customer service team, reducing the likelihood of the dataset containing malicious users, or very short term users who might be less discerning with their use of *Likes*.

Due to the sensitivity of the data and user privacy concerns, it is not possible to release the datasets used for training and testing this algorithm. However, there are no features of the dataset that would prevent this algorithm from being reproduced on similar data, and most dating services rely on binary indicators of preference such as the ones used in our algorithm.

3.2 Learning image preferences

In this section, we present a Siamese CNN [11] as a method of estimating a user's x's preference for a user y's image, based on x's history of positive and negative preferences for images.

Siamese networks have been successful in various areas such as object recognition [24] and tracking [18]. They are particularly apt at solving classification problems where the system is required to adapt to new examples quickly, known as *one-shot learning*. In the case of a classification problem, the network is trained with triplets broken into alternating pairs. The first image in the triplet, y_a is the anchor. The positive y_p is from the same classification group as the anchor, while the negative y_n is from a different group. The labels are either 1 or 0 depending on whether the inputs are (y_a, y_p) or (y_a, y_n) respectively.

The network structure is visualised in Figure 1. The symmetrical CNNs reduce the images to a 128-dimensional vector. Note that the CNN trained to create the embedding is not visualised for space concerns, but is represented in Table 1 instead. The final part of the network is then trained on the difference between the embeddings. The network is trained using a loss function that attempts to minimise the difference between the two images if they are y_a and y_p, and maximise the difference if they are y_a and y_n. This generalises the network to differentiate between images of different classes - in this case, to differentiate between an image which a user x would *Like* and an image which a user x would *Nope*. This subsequently allows us to make predictions about preference.

[5] http://dlib.net/

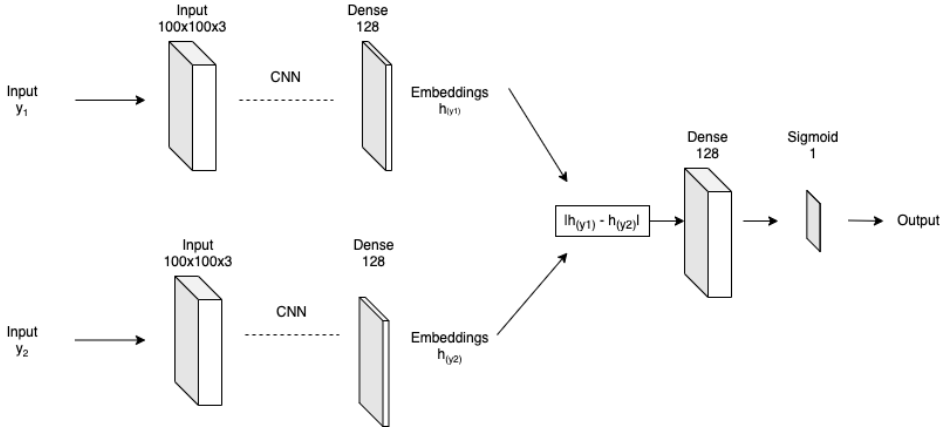


Fig. 1. Siamese network visualisation. The CNN is not not visualized for space reasons. Refer to Table 1 for the CNN architecture details.

Layer	Size-in	Size-out	Kernel	Param
input		100x100x3		0
conv1	100x100x3	100x100x3	7x7x3	444
maxpooling1	100x100x3	34x34x3	3x3	
normalization1	34x34x3	34x34x3		12
conv2	34x34x3	34x34x64	3x3x64	1792
maxpooling2	12x12x64	12x12x64	3x3	
normalization2	12x12x64	12x12x64		256
conv3	34x34x3	12x12x192	2x2x192	49344
maxpooling3	12x12x64	4x4x192	3x3	
conv4	4x4x192	4x4x384	2x2x384	295296
maxpooling4	4x4x384	2x2x384	3x3	
conv5	2x2x384	2x2x256	1x1x256	98560
conv6	2x2x256	2x2x256	3x3x256	590080
maxpooling5	2x2x256	1x1x256	3x3	
flatten	1x1x256	256		
dense1	256	256		65792
dense2	256	128		32896

Table 1. The structure of the CNN used as the symmetrical part of the network to create embeddings

Because of the abundance of data, and because the data involves complex images of varying quality, most of which contain people but from varying angles and sometimes with filters, we designed a deep CNN to create the initial embeddings. The structure of the network is described in Table 1. Because the face is likely to be an important part of a user's positive or negative reaction to other users, we used a number of layers with small convolution kernels, which has been demonstrated to be effective in face recognition and evaluation settings.

The network learns based on the difference between the outputs of the two symmetrical parts of the network via a shared weight parameter W. We use W to map y_1 and y_2 to h_{y1} and h_{y2}, which are two points in a 128 dimensional space. We can then calculate the distance between these two lower dimensional points as follows:

$$D_W(y_1, y_2) = |h_{y1} - h_{y2}| \tag{1}$$

Siamese networks are often trained with *Contrastive Loss*. The Contrastive Loss function, uses a *margin m*, and depending on the size of the margin, results in a high loss when the the network's prediction is wrong about two similar images.The Contrastive Loss function is defined as:

$$L(y_1, y_2) = (1 - Y)\frac{1}{2}(D_W(y_1, y_2))^2 + Y\frac{1}{2}(max(0, m - D_W(y_1, y_2))^2 \tag{2}$$

where Y is the binary indicator representing Like and Nope, $D_W(y_1, y_2)$ is the embedded distance between two images and m is the margin.

In many situations where siamese networks are used, the objective is to distinguish between distinct classes of items, and in this case a high error for similar objects in different classes is appropriate. In the case of preferences, this is not necessarily appropriate, as preferences are not necessarily categorical. We found that Binary Cross-Entropy, defined in Equation 3, which does not punish incorrect classifications of similar images, to be a more effective loss function.

$$L(y_1, y_2) = -(Y \log(g(D_W(y_1, y_2))) + (1 - Y) \log(1 - g(D_W(y_1, y_2)))) \tag{3}$$

where Y is the binary indicator representing Like and Nope, g is a Multi Layer Perceptron and $g(D_W(y_1, y_2))$ is the predicted probability of $D_W(y_1, y_2)$ resulting in a Like.

3.3 Recommendation Algorithm

In this section, we describe the operation of the ImRec algorithm, incorporating the model described in Section 3.2. The algorithm is visualised in Figure 2. Given two users x and y, the algorithm has four steps to calculate a bidirectional preference relation that represents the likelihood the two users will like each other.

In Step 1, the users previously *Liked* by users x and y are identified, and those users' main images are extracted. We capped the number of images used for each user at the 30 most recent. This decision was made to maintain relevance.

In Step 2, the images from Step 1, in addition to the inputs of the candidate user, are used as inputs to the appropriate siamese network. For instance, the case that x is a male user, y's image is used as the anchor (y_a), and the images x has *Liked* (y_p) are used as the positive samples while the images that x has *Noped* (y_n) are used as negative samples for the model trained on male preferences for female user images. The output is a list of scores, one for each comparison between x's image and each (y_a, y_p) pair.

The output from Step 2 is a list of scores, and Step 3 aggregates these scores into a single score. In order to do this, we first bin the scores into 5 bins of equal size between 0.0 and 1.0, and convert the bins to a distribution. We use this distribution as the input to a random forest regressor, which outputs a single value that represents a unidirectional preference score. We trained the regressor on a training set of 10000 samples independent of the training data for the siamese network. We found that this slightly outperformed simpler methods such as the Pythagorean means, and that there was no difference between the random forest and a neural network.

In Step 4, we aggregate the two unidirectional preference scores (representing x's preference for y and y's preference for x) into a single bidirectional preference score. We use the harmonic mean to aggregate the two scores. Our decision here is motivated by research indicating that the harmonic mean performs well in RRS

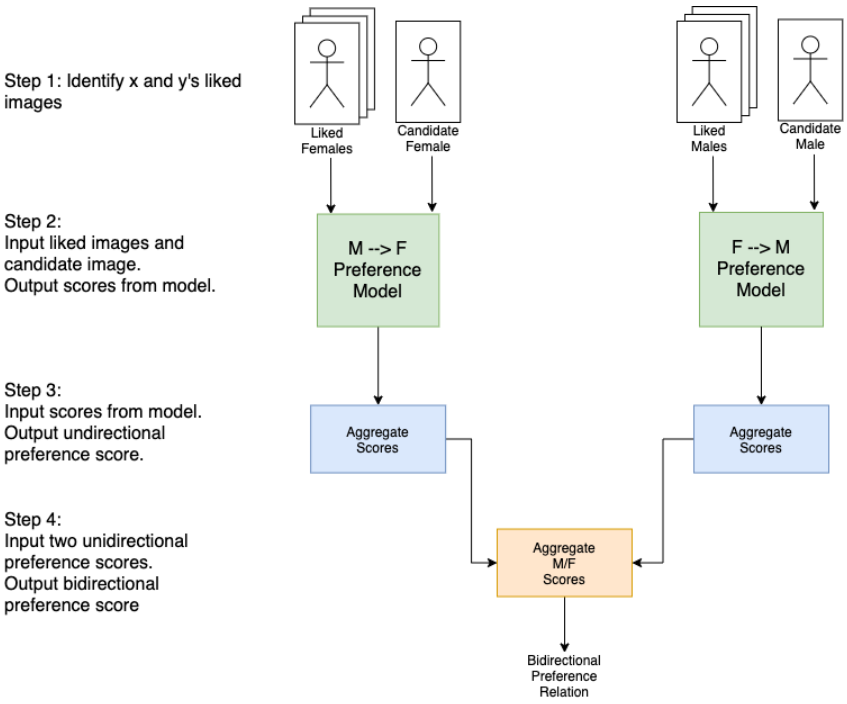


Fig. 2. ImRec visualisation

contexts [21], and also because of our desire to keep our research as consistent as possible with our baseline, RECON, which also uses the harmonic mean.

The methods in this section are tailored to matching female and male users because of the data we had available to us and because the algorithm is easier to visualise and explain with two distinct classes of users. However, the algorithm could easily be adapted to users of any orientation by creating a personalised preference model for each user with their *Liked* and *Candidate* users, including only those for whom reciprocal interest is possible based on their own orientation.

4 EVALUATION

In this section, we first describe the specifics of the dataset used in the study and the actions we took to ensure that it was representative. We then discuss the metrics we used in our evaluation, and our reasons for those choices. Finally, we report the results of the experiments we conducted to validate our model and the ImRec algorithm.

4.1 Experimental Setup and Data

The dataset for our evaluation comes from *Pairs*. Pairs is the most popular dating service in Japan and Taiwan, and growing in a number of other Asian countries. For our training data and experiments, we used only data from Japan. Data from one country is likely to be more consistent, and there are just under 10 million users registered in Japan. We also limited indications of preference to the most recent year. Over time the service user interface, search criteria and features available to paid members have changed, which may have an impact on

users' indications of preference. For the purpose of creating an accurate and representative model, these should be kept as consistent as possible.

We also made a number of other practical choices in restricting the data used in training and experiments. We limited the users used to those in the Kanto area of Japan, which includes Tokyo and the surrounding prefectures. This area represents the vast majority of Pairs users and preference indications. Users within the Kanto area are likely to decline users outside of it for practical reasons unrelated to attraction. Pairs users are also required to confirm their identity with a form of personal ID such as a driving license before they can subscribe or send messages on the service. We restricted the users we used to those who had completed this procedure. This is likely to remove many of the spammers who try to use the service for advertising or other purposes besides dating.

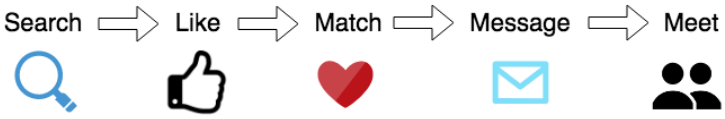


Fig. 3. The Pairs usage flow

As explained briefly in Section 3.1, users on Pairs go through a streamlined process of interaction. Users find each other using search, or from recommendations. After viewing a user's profile, a user x can choose to *Like* another y if they want to interact with them. User y is notified that they have been *Liked* and can choose to respond with a *Like* in return, or with a *Nope*. In the case that x and y *Like* each other, this is considered a *Match*. Users who *Match* can message each other, and must each send at least one message before they can exchange contact details and arrange to meet. This is visualised in Figure 3.

If we consider that meetings and relationships are the main objectives of online dating services, successful recommendations should result in these. However, because users often quickly move their communications off the site it is difficult to use this signal as a reliable indication of success or failure. We therefore evaluate our model by looking at how well it predicts *Matches*, and use successful match predictions as a proxy for a successful recommendation. Research within *Pairs* shows that users who get a large number of *Matches* are more likely to leave the service with the reason that they have found a relationship, so we consider this a reasonable proxy.

Unlike some modern dating services that emphasise fast decision-making based on the photo alone, Pairs presents the photo, categorical and textual information together before users decide to *Like* or not. We therefore feel that RECON is a fair comparison as a baseline algorithm.

4.2 Evaluation Metrics

We used offline evaluation to demonstrate the effectiveness of our algorithm. We ran cross-validation on 20000 pairs of users, 10000 of whom had *Matched* with each other, and 10000 pairs where one of the two users had sent a *Nope* to the other user. These represent successful and unsuccessful reciprocal recommendations. We balanced the test data because the number of positive and negative indicators of preference in our whole dataset is approximately equal. All of the pairs were from the past year, within the bounds explained in the previous section.

Our recommendation algorithm outputs a value between 1.0 and 0.0. In evaluating our algorithm, we define a threshold α where output values higher than α are considered predictions of *Matches* and lower values are considered predictions of *Nopes*. This allows us to calculate useful metrics such as precision and recall, and draw ROC curves that allow us to visualise the effectiveness of our algorithm across different values of α.

We focus on precision and recall as effectiveness metrics, as defined for the RRS domain by Pizzato et al. [20]. Precision is the proportion of the results that are true positive as compared to the number of false positives. In this case, *RL* is defined as the set of users who were recommended each other and *Matched* with each other, and *RN* is the set of users who were recommended to each other but at least one of them sent a *Nope*, it is defined as:

$$Precision = \frac{|RL|}{|RL| + |RN|} \tag{4}$$

Precision is considered particularly important for RSs because users lose trust in systems where a high number of unsatisfactory recommendations are displayed.

Recall is defined as the proportion of total positive results that are returned by the algorithm. A low recall indicates a small total number of recommendations, and therefore a high chance that a returning user will see the same recommendations repeatedly. Where R is the set of total recommended users, recall is defined as:

$$Recall = \frac{|RL|}{|R|} \tag{5}$$

F1 Score is a combination of precision and recall commonly used in machine learning as a measure of the overall effectiveness of the system. It is defined for RRSs in the same way as in other domains, as the harmonic mean of Precision and Recall:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

Note that success of an algorithm in the context of these metrics based on matches in a reciprocal setting implies a degree of coverage that it does not in non-reciprocal settings. Success based on matches would not be achieved by recommending the top users on the service, as these users match with a very small percentage of their recommendations, so even if it achieved one-way success, it would not generate a high precision in this evaluation. In order to ensure fairness, it would be straightforward to apply existing reciprocal methods such as [2].

4.3 Image Preference Model Results

In this section, we describe the results of the image preference prediction model. This is the siamese network described in Section 3.2 that represents Step 2 in Figure 2. To the best of our knowledge, this model is the first of its kind: there are no other models that attempt to predict personal attractiveness based on images. Because of this, we present the results for this model without a point of comparison.

The ROC curve, based on a test set of 20000 interactions from users not in the original dataset, shows that the model is capable of successfully predicting user preference based on a single image. Although the model is not always accurate in this prediction, the fact that users often *Like* a relatively large number of other users means that this model can be used as a base for predictions based on results from the model over a large number of interactions.

4.4 Results for Content-Based Algorithms

In this subsection, we present the results for ImRec compared to the current state-of-the-art content-based RRS, RECON.

Figure 5 shows the ROC curve for ImRec versus our baseline of RECON. The reference line is displayed as a dotted line. The graph was drawn using 1000 different thresholds between 0.0 and 1.0. ImRec generally has a positive and predictable curve, indicating that it is correctly predicting indicators of preference based on the users' images. On this dataset, our baseline RECON performed poorly, often worse than the reference.

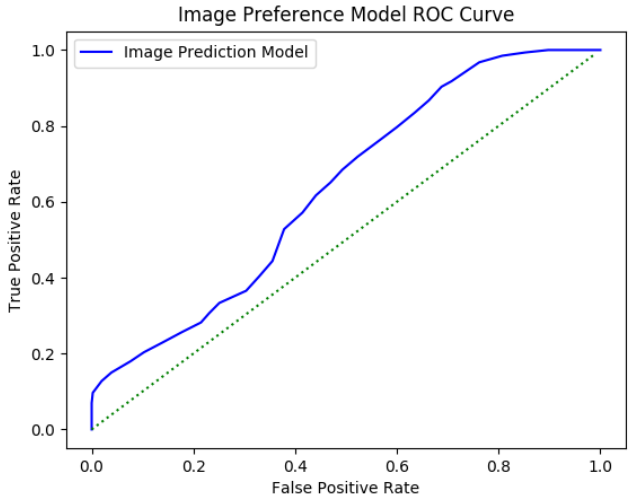


Fig. 4. ROC Curve for siamese network to predict image preferences.

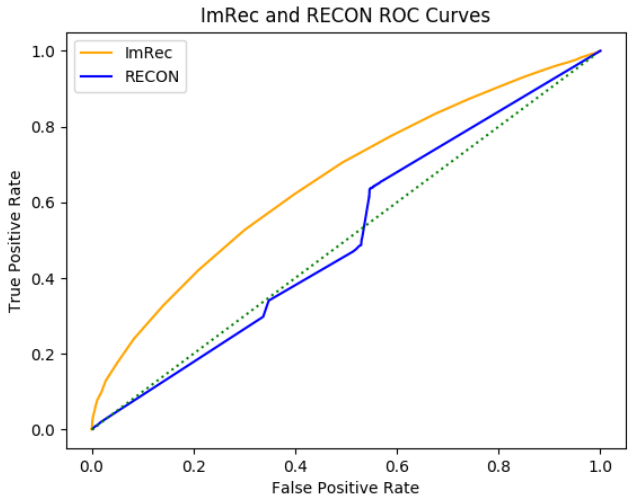
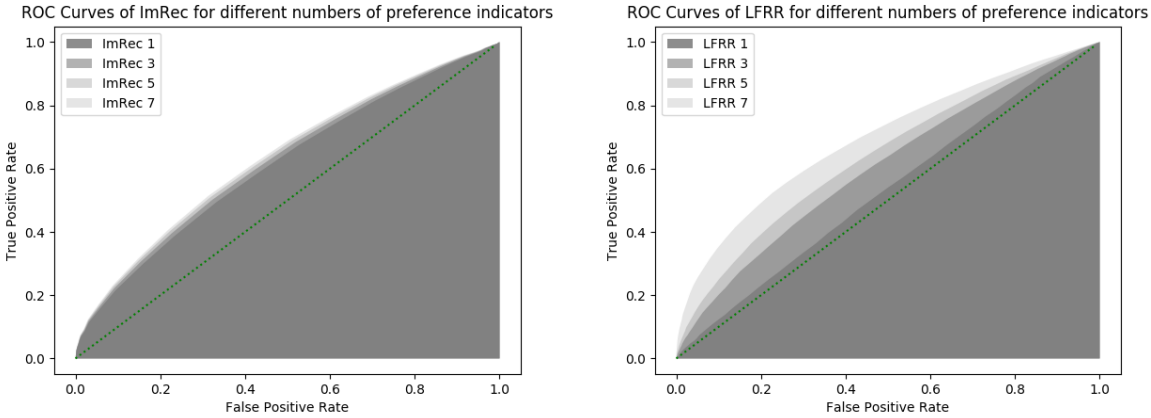


Fig. 5. ImRec and RECON ROC curves.

The main reason for RECON's poor performance on this dataset in spite of a good performance on its original test dataset is likely to be the lack of images in the dataset it was tested on. Pizzato et al. state that RECON was designed for a dataset where, "The profile of a user is made of two components: free text information and a pre-defined list of attributes, ..." [20]. In contrast, *Pairs* and many other modern online dating services and social networks use images very prominently, and users are often given an opportunity to make a positive or negative



(a) ImRec curves.

(b) LFRR curves.

Fig. 6. Curves for ImRec and LFRR for cold-start situations for various numbers of preference indicators.

decision about another user based on an image and no other information. In this situation, ImRec provides a clear advantage.

Algorithm	Precision	Recall	Best F1 Score	AUC
ImRec	0.59	0.91	0.71	0.65
RECON	0.59	0.64	0.61	0.51

Table 2. Results based on best F1 score for all relevant algorithms.

Table 2 shows the best F1 scores for the relevant algorithms. In this case, ImRec performs about 0.1 better than RECON. However, as is evident from their respective ROC curves, it is much easier to improve precision in the case of ImRec by increasing the threshold, whereas RECON performs much worse under these conditions on our dataset. Precision is vital for trust in recommender systems, as users who are shown a large proportion of recommendations that are not relevant to their interests are less likely to continue using the system.

4.5 Cold-Start Results

In this subsection, we present the results for ImRec in cold-start situations against the current state-of-the-art RRS, LFRR.

We hypothesised that ImRec would perform better than collaborative filtering algorithms in cold-start situations. We tested ImRec against the current best in class collaborative filtering algorithm, LFRR [22]. Using all available data, LFRR outperforms ImRec. However, with very little data, correlations between user preferences provide less useful information about the user's preferences than the information in the content-based model.

We tested ImRec and LFRR on a set of 20000 users interactions (10000 *Matches* and 10000 *Nopes*). From these users, we restricted interaction data available to the algorithm in training to a fixed number of interactions in order to simulate a new user. We make the assumption that new users *Like* and *Nope* in equal quantities, which in general is true. We name the algorithms trained on restricted data *Algorithm K* where K is the number of *Likes* and *Nopes* from the users in the test set available in the training set. For example *LFRR 1* is the LFRR algorithm

tested on a set of users whose training data consisted of one *Like* and one *Nope*; *ImRec 3* is the ImRec algorithm tested on a set of users whose training data consisted of three *Likes* and three *Nopes*.

Figure 6 shows the ROC curves for the ImRec and LFRR algorithms trained with restricted data. The LFRR curve is very close to random choice when trained with only one expression of preference, and improves quickly with more data. On the other hand, ImRec produces significantly better results with very little data, and improves more slowly as more examples become available.

Algorithm	**1 Indicator**	**3 Indicators**	**5 Indicators**	**7 Indicators**
ImRec	0.613	0.625	0.633	0.639
LFRR	0.530	0.604	0.639	0.696

Table 3. AUC for ImRec and LFRR for different preference indicators.

Table 3 shows the AUC for each of the algorithms trained with restricted data. It is clear from this that with fewer than 5 positive and negative indicators of preference available, ImRec outperforms LFRR, and the converse is true at 5 or more. Based on internal research done by *Pairs*, the first day of a user's interactions on a dating service is often essential, with the user deciding whether to commit to the service long term or give up based on their personal experience. As such, being able to make effective recommendations at an early stage is extremely useful for an RRS.

Based on these results, there are a number of ways that ImRec could be used to improve on the current best in class as part of a hybrid system. However, even the most simple method: a switching hybrid system that uses ImRec for recommendations up to 5 positive and negative interactions, is a clear and significant improvement.

5 CONCLUSIONS AND FUTURE WORK

5.1 Conclusions

In this paper, we developed a novel model that predicts user preference for image-based attractiveness. To the best of our knowledge, this is an entirely original contribution: there are a small number of models in the literature that predict general attractiveness [28][17], but none that predict personal preference. We conclude based on our large scale evaluation on real world data, that this model successfully differentiates between positive and negative preferences.

Using this model, we designed a novel recommender system, ImRec, that uses scores from our model to predict unidirectional, and subsequently bidirectional preferences.

Our algorithm was evaluated against RECON, with the optimizations suggested by Xia et al. [26]. We demonstrate that in the case of our dataset, RECON, considering only categorical data, did not produce better results than random selection. We consider that, based on psychological research on attractiveness and how people select prospective mates [32], this result may generalise to all dating services that use images, which is almost all modern services. We show that ImRec is able to predict user preferences based on their history of image preferences, and that it is therefore a significantly better content-based solution than RECON in this situation. The success of ImRec over RECON establishes the importance of images in online dating as compared to text-based information - a subject that would subsequently benefit from an in-depth analysis.

We also show that our system outperforms the state-of-the-art collaborative filtering solutions in the case where very little data is available, and therefore helps to solve the cold start problem. Cold start recommendations in online dating are particularly crucial, as based on data collected from *Pairs*, users often decide whether or not to commit to the service based on their experience in the first 24 hours.

5.2 Future Work

User profiles on most dating services contain two forms of unstructured data: images and freetext profiles. Based on psychological research we evaluated, we assumed that image data would be the best predictor of user preference [10][32]. However, the research provides generalised data, and it is likely that different users use different criteria to decide who they want to interact with. It is likely that many users consider text profiles in addition to images when deciding whether or not to express preference, and certain users may consider text of more importance. It would be useful to augment ImRec with a system that predicted reciprocal preference based on text profiles.

In the dataset we had available to us, an extremely high proportion of the user base came from a single ethnic group, and it was not possible to investigate the impact of the system on minority groups. However, in systems where images of people are used as the main data source, the fair treatment of all ethnic groups is a critical ethical concern. Before the system was adapted to use in other settings, it would be necessary to confirm that the system did not show bias, and fine tune the model if this was not the case.

Content-based recommender systems tend not to perform as well in most settings as collaborative filtering algorithms [25][35]. Content-based filtering algorithms therefore often form part of hybrid systems that improve on the results of the collaborative filtering algorithm, often by operating in cold start situations where they can outperform collaborative filtering techniques [4]. It would be interesting to test ImRec as part of a hybrid RRS that improves on the results of existing algorithms such as LFRR [22], for example in a *switching* hybrid system [6] that moves from content-based to collaborative reciprocal recommendation when sufficient data is available.

ACKNOWLEDGMENTS

This research has been supported by the EPSRC Doctoral Training Programme (DTP). The authors would also like to thank Eureka Inc. for providing the dataset to test our research and the resources to train the models, and in particular Mr. Shintaro Kaneko (CTO), Mr. Jun Ernesto Okumura (Data Director) and Mr. Yusuke Usui (AI Team Leader) at Eureka Inc. for their support.

REFERENCES

[1] Charu Aggarwal. 2016. *Recommender Systems: The Textbook* (1st ed.). Springer, London, England.

[2] Francesco Ricci Akiva Kleinerman, Ariel Rosenfeld and Sarit Kraus. 2018. Optimally balancing receiver and recommended users' importance in reciprocal recommender systems. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18)*. ACM, New York, NY, 131–139. https://doi.org/10.1145/3240323.3240349

[3] Sarit Kraus Akiva Kleinerman, Ariel Rosenfeld. 2018. Providing explanations for recommendations in reciprocal environments. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys '18)*. ACM, New York, NY, 22–30. https://doi.org/10.1145/3240323.3240362

[4] Lyle H. Ungar Andrew I. Schein, Alexandrin Popescul and David M. Pennock. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '02)*. ACM, New York, NY, 253–260. https://doi.org/10.1145/564376.564421 Hybrid recsys for cold start problem.

[5] Justin Basilico and Thomas Hofmann. 2004. Unifying collaborative and content-based filtering. In *Proceedings of the twenty-first international conference on Machine learning (ICML '04)*. ACM, New York, NY, 09–09. https://doi.org/10.1145/1015330.1015394 For content-based filtering references.

[6] Robin Burke. 2002. Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction* 12, 4 (Nov. 2002), 331–370. https://link.springer.com/article/10.1023 Hybrid recommender systems review.

[7] Weiping Li Zheng-Jun Zha Chenyi Lei, Dong Liu and Houqiang Li. 2016. Comparative Deep Learning of Hybrid Representations for Image Recommendations. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016)*. IEEE, 2545–2553. http://openaccess.thecvf.com/content_cvpr_2016/html/Lei_Comparative_Deep_Learning_CVPR_2016_paper.html

[8] Nikolay Sergievskiy Dmytro Mishkin and Jiri Matas. 2017. Systematic evaluation of convolution neural network advances on the Imagenet. *Computer Vision and Image Understanding* 161 (August 2017), 11–19. https://www.sciencedirect.com/science/article/pii/S1077314217300814

[9] Michael Jones Ejaz Ahmed and Tim Marks. 2015. An Improved Deep Learning Architecture for Person Re-Identification. In *Proceedings of the 2015 Conference on Computer Vision and Pattern Recognition (CVPR 2015)*. IEEE, 3908–3916. http://openaccess.thecvf.com/content_cvpr_2015/html/Ahmed_An_Improved_Deep_2015_CVPR_paper.html

[10] Man-Ling Fen. 2005. Choosing online partners in the virtual world: How online partners' characteristics affect online dating. *ProQuest Dissertations Publishing* (2005). https://search.proquest.com/docview/305373750/fulltextPDF/170796CB28C74552PQ/1?accountid=9730

[11] R Zemel G Koch and R Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *Proceedings of the 2015 ICML Deep Learning workshop*. ICML.

[12] Jianming He and Wesley Chu. 2010. A Social Network-Based Recommender System (SNRS). *Data Mining for Social Network Data* 12 (May 2010), 47–74. https://link.springer.com/chapter/10.1007/978-1-4419-6287-4_4

[13] A.Hernando J.Bobadilla, F.Ortega and A.Gutiérrez. 2013. Recommender Systems Survey. *Knowledge-Based Systems* 46 (2013), 109–132. https://www.sciencedirect.com/science/article/abs/pii/S0950705113001044

[14] Richard Socher Li-Jia Li Kai Li Jia Deng, Wei Dong and Li Fei-Fei. 2009. ImageNet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CCVPR 2009)*. IEEE, Miami, FL, 248–255. https://doi.org/10.1109/CVPR.2009.5206848

[15] Min-Yen Kan Jovian Lin, Kazunari Sugiyama and Tat-Seng Chua. 2013. Addressing cold-start in app recommendation: latent user models constructed from twitter followers. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval (SIGIR '13)*. ACM, New York, NY, 283–292. https://doi.org/10.1145/2484028.2484035

[16] Ken Lang. 1995. NewsWeeder: Learning to Filter Netnews. In *Proceedings 12th International Conference on Machine Learning, (ICML 1995)*. 331–339. https://pdfs.semanticscholar.org/26fd/e7f657b41be65a0b975615508f4f100e3a04.pdf

[17] Jinhai Xiang Lu Xu and Xiaohui Yuan. 2018. Transferring Rich Deep Features for Facial Beauty Prediction. *arXiv* (March 2018). https://arxiv.org/abs/1803.07253

[18] João Henriques Andrea Vedaldi Luca Bertinetto, Jack Valmadre and Philip Torr. 2016. Fully-Convolutional Siamese Networks for Object Tracking. In *Proceedings of the 2016 European Conference on Computer Vision (ECCV 2016)*. Springer, 850–865. https://link.springer.com/chapter/10.1007/978-3-319-48881-3_56

[19] Joshua Akehurst Irena Koprinska Kalina Yacef Luiz Pizzato, Tomasz Rej and Judy Kay. 2013. Recommending people to people: the nature of reciprocal recommenders with a case study in online dating. *User Model User-Adap Inter* 23, 5 (Nov. 2013), 447–488. https://link.springer.com/article/10.1007/s11257-012-9125-0

[20] Thomas Chung Irena Koprinska Luiz Pizzato, Tomek Rej and Judy Kay. 2010. RECON: a reciprocal recommender for online dating. In *Proceedings of the fourth ACM conference on Recommender systems (RecSys '10)*. ACM, New York, NY, 207–214. https://doi.org/10.1145/1864708.1864747

[21] James Neve and Ivan Palomares. 2018. Aggregation Strategies in User-to-User Reciprocal Recommender Systems. In *Proceedings of the 2018 IEEE International Conference on Systems, Man and Cybernetics (IEEE SMC 2018)*. IEEE, New York, NY, 2299–2304. Reciprocal recommender systems aggregation.

[22] James Neve and Ivan Palomares. 2019. Latent Factor Models and Aggregation Operators for Collaborative Filtering in Reciprocal Recommender Systems. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys '19)*. ACM, New York, NY.

[23] James Neve and Ivan Palomares. 2020. Hybrid Reciprocal Recommender Systems: Integrating Item-to-User Principles in Reciprocal Recommendation. In *4th International Workshop on Mining Actionable Insights from Social Networks (WebConf 2020)*. Hybrid Reciprocal Recsys.

[24] Timothy Lillicrap Koray Kavukcuoglu Oriol Vinyals, Charles Blundell and Daan Wierstra. 2016. Matching Networks for One Shot Learning. *Advances in Neural Information Processing Systems* 29 (2016). http://papers.nips.cc/paper/6068-learning-feed-forward-one-shot-learners.pdf

[25] Yehuda Koren Paolo Cremonesi and Roberto Turrin. 2010. Performance of recommender algorithms on top-n recommendation tasks. In *Proceedings of the fourth ACM conference on Recommender systems (RecSys '10)*. ACM, New York, NY, 39–46. https://doi.org/10.1145/1864708.1864721 Recommender systems evaluation real time.

[26] Yizhou Sun Peng Xia, Benyuan Liu and Cindy Chen. 2015. Reciprocal Recommendation System for Online Dating. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM '15)*. ACM, New York, NY, 234–241.

[27] Shu Wu Qiang Liu and Liang Wang. 2017. DeepStyle: Learning User Preferences for Visual Recommendation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2017)*. ACM, New York, NY, 841–844. https://dl.acm.org/doi/abs/10.1145/3077136.3080658

[28] Alireza Tavakoli Targhi Rashidedin Jahandideh and Maryam Tahmasbi. 2018. Physical Attribute Prediction Using Deep Residual Neural Networks. *arXiv* (December 2018). https://arxiv.org/abs/1812.07857

[29] David Belanger Trapit Bansal and Andrew McCallum. 2016. Ask the GRU: Multi-task Learning for Deep Text Recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems (Recsys 2016)*. ACM, New York, NY, 107–114. https://doi.org/10.1145/2959100.2959180

[30] Robin van Meteren and Maarten van Someren. 2000. Using Content-Based Filtering for Recommendation. In *Proceedings of the ECML 2000 Workshop: Maching Learning in Information Age, (ECML 2000)*. 47–56. http://users.ics.forth.gr/~potamias/mlnia/paper_6.pdf

[31] Huang Wang Wenxing Hong, Siting Zheng and Jianchao Shi. 2013. A Job Recommender System Based on User Clustering. *Journal of Computers* 8, 8 (Aug. 2013), 1960–1967.

[32] Crystal Wotipka and Andrew High. 2016. An idealized self or the real me? Predicting attraction to online dating profiles using selective self-presentation and warranting. *Communication Monographs* 83, 3 (July 2016), 281–302. https://www.tandfonline.com/doi/full/10.1080/03637751.2016.1198041

[33] Trong Duc Le Xuan Nhat Lam, Thuc Vu and Anh Duc Duong. 2008. Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication (ICUIMC '08)*. ACM, New York, NY, 208–211. https://doi.org/10.1145/1352793.1352837

[34] Sukrit Shankar Yoad Lewenberg, Yoram Bachrach and Antonio Criminisi. 2016. Predicting Personal Traits from Facial Images Using Convolutional Neural Networks Augmented with Facial Landmark Information. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence (AAAI 2016)*. AAAI, 4365–4366. https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/viewPaper/12384

[35] Robert Schreiber Yunhong Zhou, Dennis Wilkinson and Rong Pan. 2008. Large-Scale Parallel Collaborative Filtering for the Netflix Prize. In *Proceedings of the 4th international conference on Algorithmic Aspects in Information and Management (AAIM '08)*. Springer-Verlag, Berlin, Heidelberg, 337–348.

[36] Zhang Ning Zheng Siting, Hong Wenxing and Yang Fan. 2012. Job recommender systems: A survey. In *Proceedings of the 7th International Conference on Computer Science & Education (ICCSE '12)*. IEEE, Melbourne, VIC, Australia, 920–924. https://doi.org/10.1109/ICCSE.2012.6295216